OBJECT ORIENTED PROGRAMMING WITH JAVA 8– LAB

1. Create a TreeMap and add some entries to it. Display the map contents using Iterator. Check whether a particular key exists in the map or not. If it is present, display its value.

Ans=

Code-

```
import java.util.Iterator;
import java.util.Map;
import java.util.TreeMap;

public class TreeMapAdd
        {
         public static void main(String[] args)
        {
         TreeMap<String, Integer> TMAdd = new TreeMap<>();

         TMAdd.put("Sunil", 25);
         TMAdd.put("Rajesh", 27);
         TMAdd.put("Yogesh", 26);

         Iterator<Map.Entry<String, Integer>> iterator = TMAdd.entrySet().iterator();
         System.out.println("TreeMap Contents:");

         while (iterator.hasNext())
                {
                 Map.Entry<String, Integer> entry = iterator.next();
                 System.out.println(entry.getKey() + ": " + entry.getValue());
                }

         String keyToCheck = "Sunil";
         if (TMAdd.containsKey(keyToCheck))
                {
                 int value = TMAdd.get(keyToCheck);
                 System.out.println(keyToCheck + " is present in the map with a value of " + value);
                }
                else
                {
                 System.out.println(keyToCheck + " is not found in the map.");
                }
        }
}
```

Execution-

```
C:\Users\p7pha\OneDrive\Desktop\Cdac DBDA\JAVA>javac TreeMapAdd.java

C:\Users\p7pha\OneDrive\Desktop\Cdac DBDA\JAVA>java TreeMapAdd
TreeMap Contents:
Rajesh: 27
Sunil: 25
Yogesh: 26
Sunil is present in the map with a value of 25

C:\Users\p7pha\OneDrive\Desktop\Cdac DBDA\JAVA>
```

2. Create a class with a generic method to find the largest element in an array and its position.

Ans=

Code-

```
public class LargestArray
{
         public static <T extends Comparable<T>> void findLargestElement(T[] arr)
        {
         int maxIndex = 0;
         for (int i = 1; i < arr.length; i++)
                {
                 if (arr[i].compareTo(arr[maxIndex]) > 0)
                        {
                         maxIndex = i;
                        }
                }
         System.out.println("The largest element in the array is " + arr[maxIndex] + " and its position is " + maxIndex);
        }

        public static void main(String[] args)
        {
         Integer[] intArr = {1, 2, 3, 4, 5};
         String[] strArr = {"apple", "banana", "orange", "pear"};
         Character[] charArr = {'a', 'b', 'c', 'd'};

         findLargestElement(intArr);
         findLargestElement(strArr);
         findLargestElement(charArr);
        }
}
```

Execution-

```
C:\Users\p7pha\OneDrive\Desktop\Cdac DBDA\JAVA>javac LargestArray.java

C:\Users\p7pha\OneDrive\Desktop\Cdac DBDA\JAVA>java LargestArray
The largest element in the array is 5 and its position is 4
The largest element in the array is pear and its position is 3
The largest element in the array is d and its position is 3

C:\Users\p7pha\OneDrive\Desktop\Cdac DBDA\JAVA>
```

3. Create an Employee class with data members empid, first name, last name, dept and salary. Create a Treeset of Employee objects and sort objects using first name. If two employees have the same first name, then sort them by last name using Comparator.

Ans=

Code-Part1,

```
import java.util.*;

class Employee
{
        int empid;
        String firstName;
        String lastName;
        String dept;
        double salary;

        public Employee(int empid, String firstName, String lastName, String dept, double salary)
        {
         this.empid = empid;
         this.firstName = firstName;
         this.lastName = lastName;
         this.dept = dept;
         this.salary = salary;
        }

        public int getEmpid()
        {
         return empid;
        }

        public String getFirstName()
        {
        return firstName;
        }

        public String getLastName()
        {
        return lastName;
        }

        public String getDept()
        {
        return dept;
        }

        public double getSalary()
        {
        return salary;
        }
```

Part2,

```
        @Override
        public String toString()
        {
         return "Employee{" +
                                "empid=" + empid +
                                ", firstName='" + firstName + '\'' +
                                ", lastName='" + lastName + '\'' +
                                ", dept='" + dept + '\'' +
                                ", salary=" + salary +
                        '}';
        }
}

public class EmployeeDetails
        {
        public static void main(String[] args)
        {
        TreeSet<Employee> employeeSet = new TreeSet<>(new EmployeeComparator());

        employeeSet.add(new Employee(101, "Sunil", "Shahi", "HR", 55000.0));
        employeeSet.add(new Employee(102, "Rushi", "Tapdiya", "IT", 60000.0));
        employeeSet.add(new Employee(103, "Pramod", "Bhombe", "Finance", 52000.0));
        employeeSet.add(new Employee(104, "Mahesh", "Gadekar", "IT", 62000.0));

        for (Employee employee : employeeSet)
                {
                 System.out.println(employee);
                }
        }
}
```

Part3,

```
class EmployeeComparator implements Comparator<Employee>
{
        @Override
        public int compare(Employee emp1, Employee emp2)
               {
               int firstNameComparison = emp1.getFirstName().compareTo(emp2.getFirstName());
               if (firstNameComparison != 0)
                       {
                        return firstNameComparison;
                       }

        int lastNameComparison = emp1.getLastName().compareTo(emp2.getLastName());
        if (lastNameComparison != 0)
                       {
                        return lastNameComparison;
                       }

       return Integer.compare(emp1.getEmpid(), emp2.getEmpid());
       }
}
```

Execution-

```
C:\Users\p7pha\OneDrive\Desktop\Cdac DBDA\JAVA>javac EmployeeDetails.java

C:\Users\p7pha\OneDrive\Desktop\Cdac DBDA\JAVA>java EmployeeDetails
Employee{empid=104, firstName='Mahesh', lastName='Gadekar', dept='IT', salary=62000.0}
Employee{empid=103, firstName='Pramod', lastName='Bhombe', dept='Finance', salary=52000.0}
Employee{empid=102, firstName='Rushi', lastName='Tapdiya', dept='IT', salary=60000.0}
Employee{empid=101, firstName='Sunil', lastName='Shahi', dept='HR', salary=55000.0}

C:\Users\p7pha\OneDrive\Desktop\Cdac DBDA\JAVA>
```